

LAMK Lahden ammattikorkeakoulu
Lahti University of Applied Sciences

Taloyhtiöiden asiakirjapalvelu

LAHDEN
AMMATTIKORKEAKOULU
Tekniikan ala
Tietotekniikka
Ohjelmistotekniikka
Opinnäytetyö
Kevät 2015
Mikko Pullinen

Lahden ammattikorkeakoulu
Tietotekniikka

PULLINEN, MIKKO:

Taloyhtiöiden asiakirjapalvelu

Ohjelmistotekniikan opinnäytetyö, 35 sivua

Kevät 2015

TIIVISTELMÄ

Tämä työ toteutettiin Ohjelmistotalo Koodiavaimelle. Työssä suunniteltiin ja toteutettiin palvelu vastaamaan taloyhtiöiden asiakirjatarpeisiin. Palvelun tarkoituksena on tarjota keskitetty vaihtoehto taloyhtiöiden erilaisiin asiakirja- ja tiedoitustarpeisiin.

Taloyhtiöiden asiakastarpeita kartoitettiin yhdessä projektiin osallistuvien sekä potentiaalisten ja pilottiasiakkaiden kanssa. Keskeisemmäksi vaatimukseksi nousi helppokäyttöinen keskitetty sähköinen vaihtoehto erilaisten asiakirjojen ja tiedotteiden laadintaan, arkistointiin ja jakamiseen asiakkaiden ja sidosryhmien välillä.

Palvelu toteutettiin käyttämällä useita web-teknologioita. Palvelun toteutuksessa käytettiin pääsääntöisesti PHP- ja JavaScript-ohjelmointikieliä. Näiden lisäksi käytettiin myös jQuery ja jQuery UI -JavaScript-ohjelmointikehyksiä. Palvelu käyttää myös Wkhtmltopdf-ohjelmistoa PDF-tiedostojen generoinnissa HTML-kuvauskielestä. Palvelu toteutettiin Concrete5-sisällönhallintajärjestelmän päälle. Www-käyttöliittymä toteutettiin HTML5-kuvauskielellä ja CSS 3.0 -tyyliohjeilla. Palvelun toteutus hyödyntää Concrete5:n tarjoamaa MVC-sovellusarkkitehtuuria. MVC-sovellusarkkitehtuurin näkymien ja ohjaimien välinen kommunikointi on hoidettu POST-tyyppisillä viesteillä tai AJAX-kutsuilla.

Työn tuloksena valmistui palvelu, joka tarjoaa taloyhtiöille keskitetyn vaihtoehdon heidän asiakirjatarpeisiin. Palvelu on pilottivaiheessa, ja asiakkaiden kokemukset ja mielipiteet otetaan huomioon palvelua jatkokehitettäessä.

Asiasanat: taloyhtiö, asiakirja, Concrete5, JavaScript, jQuery, jQuery UI

Lahti University of Applied Sciences

Degree Programme in Information Technology

PULLINEN, MIKKO:

Document service of housing
cooperatives

Bachelor's Thesis in Software Engineering, 35 pages

Spring 2015

ABSTRACT

This Bachelor's thesis was done for Ohjelmistotalo Koodiavain. This work deals with designing and developing a service which satisfies the documenting needs of housing cooperatives. The goal of the work was to offer an alternative for various documenting and informing needs of housing cooperatives.

The needs of housing cooperatives were surveyed with the participants of this project, potential and pilot customers of the service. Having a centralized easy-to-use electronic alternative of document creation, archiving and sharing between the clients and stakeholders rose as the main requirement for the service.

The service was implemented by using several web technologies. The service was mainly implemented by using PHP and JavaScript programming languages. In addition, jQuery and jQuery UI JavaScript frameworks were used. The service also uses the Wkhtmltopdf program in order to generate PDF documents out of HTML markup language. The service was implemented on top of the Concrete5 content management system. The www user interface was implemented with HTML5 markup language and CSS 3.0 style sheet language. The service uses Concrete5's MVC software architectural pattern. The dataflow between views and controllers of the MVC software architectural pattern is implemented with either POST type messages or AJAX calls.

The service that was developed offers a centralized alternative for the documenting needs of housing cooperatives. The service is currently in the pilot stage. The opinions and suggestions of customers are being listened to and taken into consideration, for future development of the service.

Key words: housing cooperative, document, Concrete5, JavaScript, jQuery, jQuery UI

LYHENNELUETTELO

AJAX	Asynchronous JavaScript And XML
CSS	Cascading Style Sheets
DPI	Dots per inch
HTML	Hypertext Markup Language
PHP	PHP: Hypertext Preprocessor
MVC	Model-view-controller
URL	Uniform resource locator

SISÄLLYS

1	JOHDANTO	1
2	ASIAKKAAT JA ASIAKASTARVE	2
3	PALVELUN KUVAUS	4
4	TYÖKALUN KUVAUS	7
5	WWW-TEKNOLOGIAT	14
5.1	MVC-arkkitehtuuri	14
5.2	HTML, CSS, PHP ja JavaScript	15
5.3	jQuery, jQuery UI ja AJAX	16
6	CONCRETE5-SISÄLLÖNHALLINTAJÄRJESTELMÄ	18
6.1	Käyttöönotto	18
6.2	Rakenne	18
6.3	Ohjaimet	19
6.4	Mallit	20
6.5	Näkymät	20
6.5.1	Yksittäiset sivut	20
6.5.2	Sivutyypit	21
6.6	Teemat	22
6.7	Työkalut	23
6.8	Paketit	23
7	PALVELUN TOTEUTUS	25
7.1	Vaatimusten kartoitus	25
7.2	Yksittäiset sivut	25
7.3	Ohjaimet	26
7.4	Mallit	26
7.5	Teema	27
7.6	Työkalun toteutus	28
7.7	Hallintasivu toteutus	30
7.8	Toteutuksen yhteenveto	33
8	YHTEENVETO	35
	LÄHTEET	36

1 JOHDANTO

Taloyhtiöiden asiakirjatarpeisiin kuuluu niiden luonti, muokkaaminen, helppo hallinta ja jakaminen. Tarpeiden täyttämiseksi usein käytetään useampaa ohjelmaa. On tärkeää, että taloyhtiöiden talojen tiedot ovat ajantasalla ja helposti saatavilla. Asiakirjoja ja tietoja jaetaan taloyhtiöissä hallituksen, asukkaiden ja sidosryhmien kesken.

Suomen Talotiedot on Ohjelmistotalo Koodiavaimen, Omakirja Oy:n ja Mikko Pullisen projekti, jossa kehitettävä palvelu tarjoaa pilvipalveluna toteutetun vaihtoehdon taloyhtiöiden asiakirjatarpeisiin. Suomen Talotiedot Oy on perustettu vuonna 2014. Yrityksen toimiala on ohjelmistojen suunnittelu ja valmistus.

Työn tavoitteena on toteuttaa helppokäyttöinen palvelu, joka vastaa taloyhtiöiden kaikkiin asiakirjatarpeisiin. Asiakaspalautetta kerätään pilottiasiakkaiden sekä maksavien asiakkaiden palautteiden kautta. Asiakaspalautteet otetaan huomioon palvelua kehitettäessä. Työssä tutustutaan taloyhtiöiden tarpeisiin sekä näiden tarpeiden vastaamiseen vaadittaviin web-teknologioihin.

Tämän työn luvussa 2 käsitellään taloyhtiöiden asikastarpeista. Luvussa 3 ja 4 käsitellään tarjottavaa palvelua. Luku 5 ja 6 käsittelevät palvelussa käytettyjä web-teknologioita. Luku 7 käsittelee palvelun käytännön toteutusta.

2 ASIAKKAAT JA ASIAKASTARVE

Palvelun vaatimuksia kartoitettiin projektin alkuvaiheessa projektiin osallistuvilta osapuolilta ja heidän tuntemiltaan henkilöiltä. Potentiaalisilta asiakkailta saadut mielipiteet täsmensivät palvelun suunnan palvelemaan taloyhtiöiden tarpeita. Taloyhtiöiden keskeisemmäksi vaatimukseksi nousi helppokäyttöinen keskitetty sähköinen vaihtoehto erilaisten asiakirjojen ja tiedotteiden laadintaan, arkistointiin ja jakamiseen asiakkaiden ja sidosryhmien välillä.

Taloyhtiön hallituksen puheenjohtaja haluaa, että hänen taloyhtiönsä perustiedot, isännöitsijän, huoltoyhtiön ja hallituksen yhteystiedot sekä vastuunjakotaulukko ovat helposti löydettävissä ja kaikkien asianomaisten saatavilla. Lisäksi hän toivoo, että tietoihin on pääsy mobiilialustoilta. Kaikki taloyhtiöön liittyvät asiakirjat ja -paperit tulevat olla aina asianomaisten saatavilla ja löydettävissä helposti samasta paikkaa. Taloyhtiöön liittyvät tiedot tulevat olla myös päivitettävissä ja päivityksen näkyä reaaliajassa kaikille palvelun asianomaisille käyttäjille.

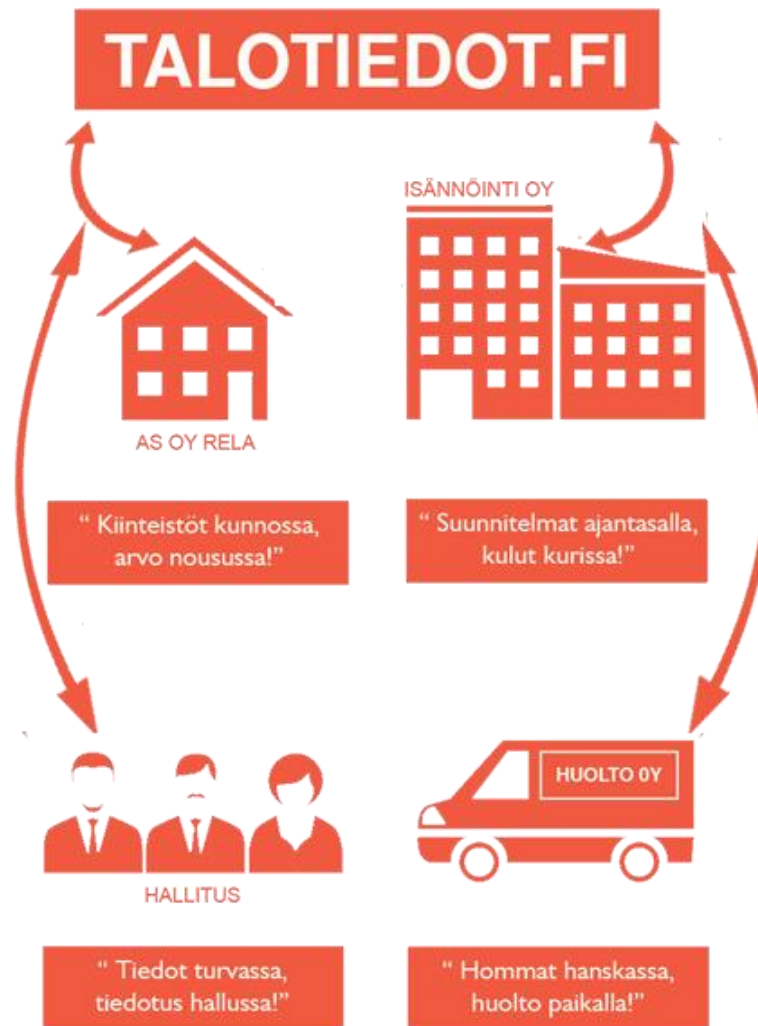
Palvelun tulee olla selkeä, yksinkertainen ja käyttäjälähtöinen, jotta kaikki palvelun käyttäjät ovat kykeneviä käyttämään palvelua. Palvelun käytön helpottamiseksi palvelusta tulee löytyä hakutoiminto. Palvelussa säilytettävien tietojen tulee olla helposti luettavissa käytettävän laitteen ruudulta. Monisivuisten asiakirjojen sivuja pitää pystyä tulostamaan myös yksittäin. Asiakirjoja luotaessa niihin pitää pystyä tuomaan aiemmin luoduista ja arkistoiduista PDF-asiakirjoista joko useita tai yksittäisiä sivuja. Valmis asiakirja tulee pystyä tilaamaan fyysisenä kirjana.

Taloyhtiöiden asioista tiedottaminen tulee olla mahdollista palvelun kautta. Tiedottamisen helpottamiseksi palvelusta tulee löytyä valmiita asiakirja- ja tiedotepohjia ja niistä löytyvien yhteystietojen tulee päivittyä automaattisesti. Kun palvelussa on luotu omia asiakirjoja ja tiedotteita, tulee niitä pystyä muokkaamaan. Palveluun tulee myös pystyä luomaan omia asiakirjapohjia, joita voi käyttää samoin tavoin kuin palvelun tarjoamia valmiita asiakirja- ja tiedotepohjia. Asiakirjojen ja asiakirjapohjien tulee tukea kiinteistöliiton ja isännöitsijäliiton tekemiä ohjeita. Palvelu tulee kerätä lokia siitä, kenelle ja mitä

tiedotteita on lähetetty. Kun tiedote on vastaanotettu, tulee se pystyä kuitata vastaanotetuksi. Taloyhtiöiden tulee pystyä määrittelemään käyttäjätasoja taloyhtiön käyttäjille. Määritettyjä käyttäjäryhmiä tulee pystyä käyttämään taloyhtiöiden asiakirjojen jakamisen kohdistamiseen ja tietojen saatavuuden rajoittamiseen.

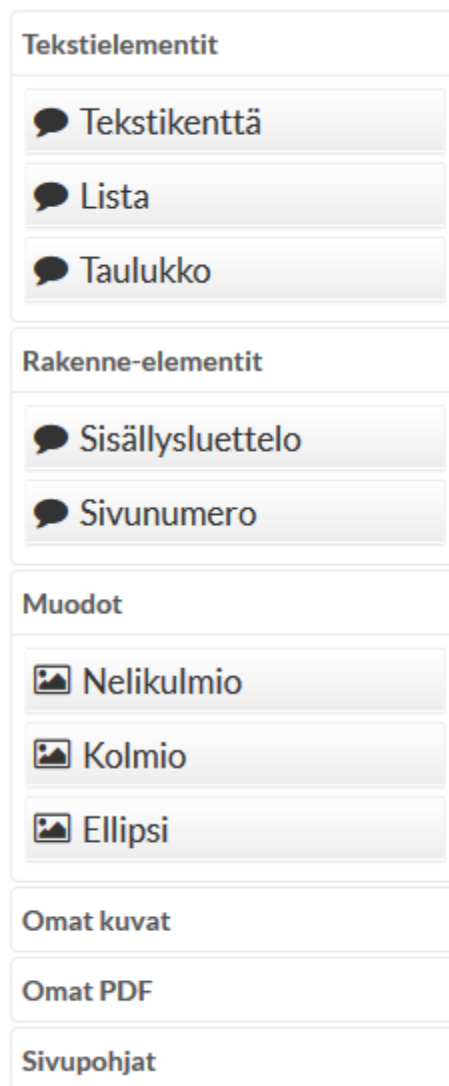
3 PALVELUN KUVAUS

Palvelu tarjoaa mahdollisuuden taloyhtiöiden tiedotus- ja asiakirjatarpeiden keskittämisen yhteen paikkaan. Talonyhtiöiden asiakastarpeisiin kuuluu kokouksien pöytäkirjojen jakaminen helposti kaikille asukkaille, tiedostojen jakaminen ja arkistointi tukiverkoston kuten isännöitsijöiden kanssa ja helppokäyttöinen pilvipalveluna toteutettu tekstinkäsittelytyökalu, joka antaa mahdollisuuden luopua ylinmääräisistä työpöytäympäristön tekstinkäsittely-ohjelmistoista. Pavelun tekstinkäsittelyohjelma ja omien tietojen hallinta tulee toteuttaa mahdollisimman helppokäyttöisesti asiakaskunnan keskittyessä ei-tietotekniselle alalle. Kuviossa 1 on perusidea palvelusta, sen käyttäjistä ja tarpeista.



KUVIO 1. Asiakastarve ja ratkaisu

Työkalussa käyttäjä haluaa laatia asiakirjoja raahaa ja pudota -menetelmällä asettaen haluamiaan työkalun tarjoamia muokattavia elementtejä. Työkalun tulee tarjota käyttäjälle laaja valikoima erilaisia elementtejä asiakirjojen laadintaa varten. Kuviossa 2 esitellään palveluun tarvittavia elementtejä. Käyttäjällä tulee olla mahdollisuus myös tuoda omia aiemmin luomiaan kuvia ja PDF-tiedostoja käytettäväksi palveluun oman yrityksensä sisälle joko henkilökohtaiseen tai yleiseen käyttöön.



KUVIO 2. Työkalun tarjoamat elementit

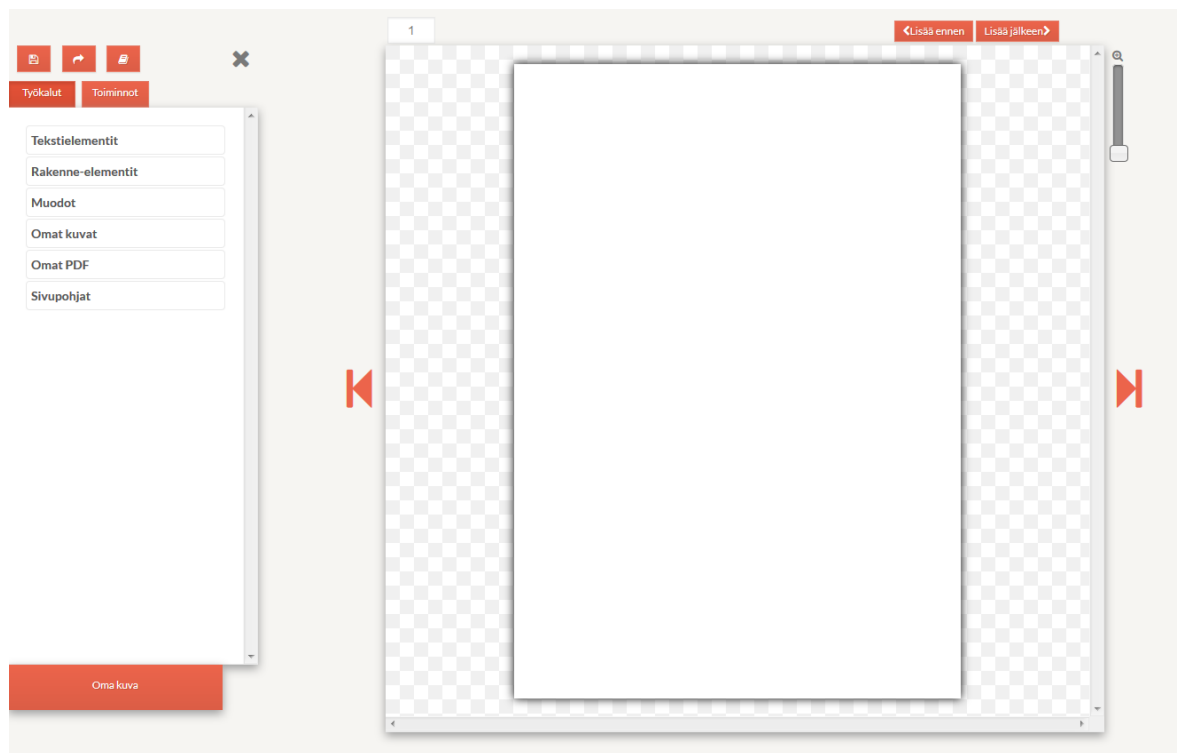
Palvelun hallintasivu tulee olla näkyvissä vain käyttäjille, jotka ovat aktivoineet yhden neljästä eri käyttäjätasosta tai ovat saaneet ylläpitäjäoikeudet johonkin

yritykseen. Ilmaisen käyttäjätason tulee mahdollistaa palvelun peruskäytön. Ilmaisella käyttäjätasolla tulee olla rajattu määrä yrityksiä, muokkaus-oikeuksellisia asiakaskäyttäjätilejä ja käytettävää levytilaa. Maksullisten käyttäjätasovaihtoehtojen tulee tarjota enemmän yrityksiä, muokkaus-oikeuksellisia käyttäjiä ja levytilaa. Käyttäjätasot on tarkoitettu pääsääntöisesti käyttäjille, jotka ovat halukkaita luomaan omille yrityksillensä portaaleja palveluun. Käyttäjät, jotka lisätään yritykseen joko pelkällä lukuoikeudella tai sekä muokkaamisoikeudella, eivät tarvitse välttämättä käyttäjätason suomaan oikeutta luoda omien yrityksiensä portaaleja palveluun. Jokaisella käyttäjällä tulee kuitenkin olla mahdollisuus aktivoida yksi saatavilla olevista käyttäjätasoista tarpeen vaatiessa.

Hallintasivulla yrityksen omistajakäyttäjä tai ylläpitäjäoikeuden omaavan käyttäjän tulee voida hallinnoida yrityksen käyttäjäryhmiä, käyttäjiä ja yrityksen tiedostoja. Hallintasivun tulee tarjota tarvittavat ominaisuudet yrityksen tietojen hallintaan ja palvelussa työskentelyn mahdollistamiseksi kaikille asianomaisille käyttäjille.

4 TYÖKALUN KUVAUS

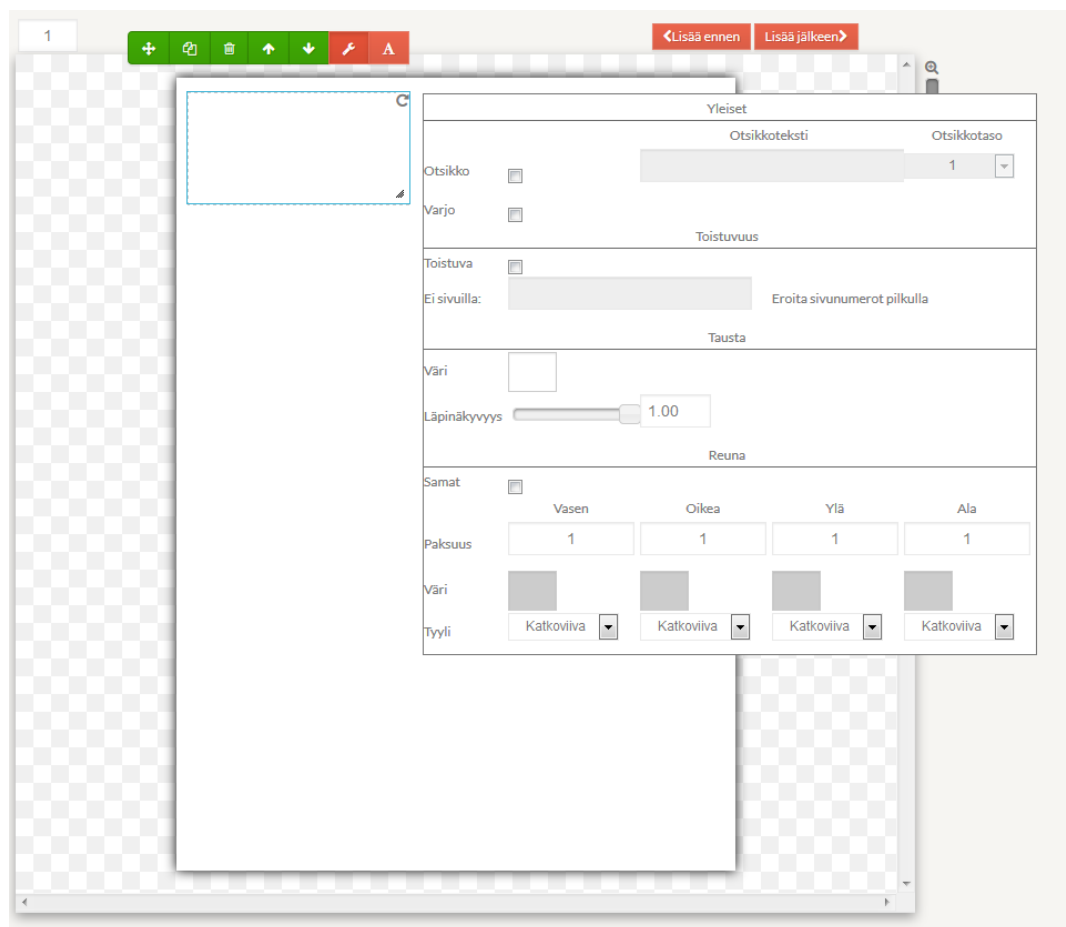
Palvelun tekstinkäsittelytyökalu on toteutettu käyttäen JavaScriptiä, jQueryä ja jQuery UI:ta. Kaikki asiakirjaan tehtävät muutokset tehdään asiakkaan selaimessa. Muutokset eivät tule voimaan asiakirjaan reaaliaikaisesti. Muutokset näkyvät muille käyttäjille vasta, kun muutokset tallennetaan. Työkalu tarjoaa käyttäjälle valikoiman raahaa ja pudota -menetelmällä asetettavia rakennuselementtejä asiakirjojen laatimiseen. Käyttäjä voi myös tuoda omia kuviaan ja PDF-tiedostoja palveluun käytettäväksi asiakirjojen laadinnassa. Kuviossa 3 on kuvakaappaus työkalusta.



KUVIO 3. Työkalu

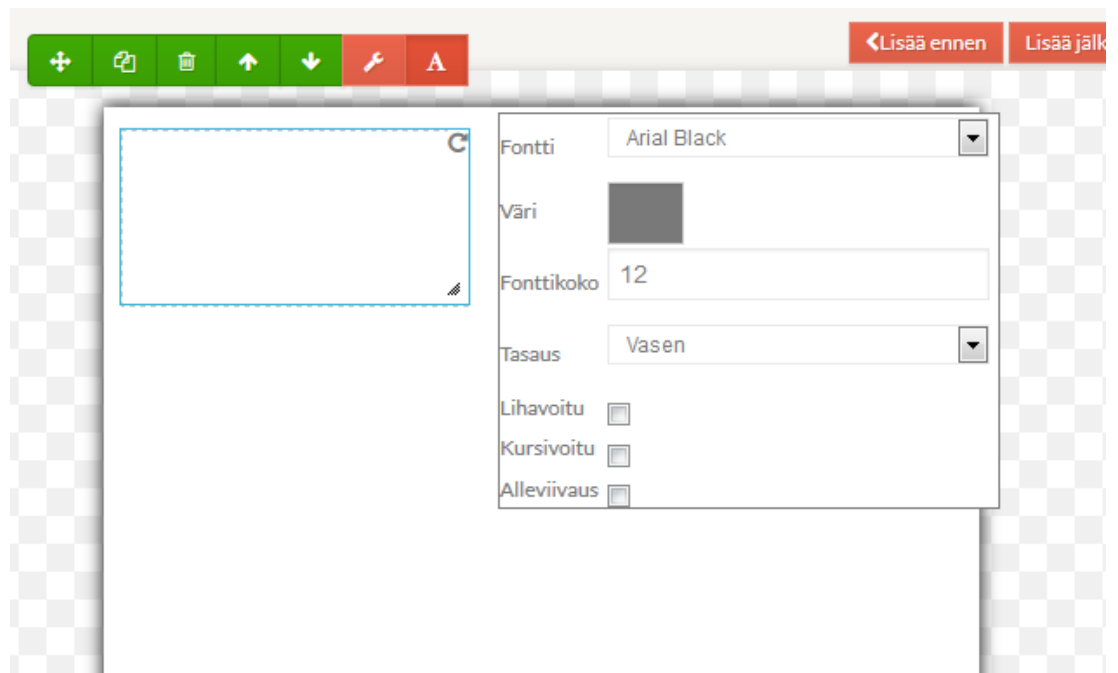
Työkalun elementtien kokoa, sijaintia ja kiertoa voidaan muokata helposti. Käyttäjän valittua elementti voidaan sitä siirtää siirtotyökalulla raahaamalla. Elementin oikeasta alakulmasta löytyy koon muuttamiseen käytettävä raahattava työkalu. Elementin oikeasta yläkulmasta löytyy kierron muuttamiseen käytettävä raahattava työkalu. Kaikille elementeille yhteisiä työkaluja ovat myös monistaminen, poistaminen, eteen tuonti ja taakse vienti.

Työkalun elementtien perusasetuksista voidaan asettaa eri elementtejä otsikoiksi, jolloin ne ilmestyvät mahdollisesti asetettuun sisällysluetteloelementtiin automaattisesti sivunumeroituna. Elementin otsikkotasoa voidaan myös muuttaa asetuksista halutuksi. Jokaiselle elementille on mahdollista asettaa varjo. Palvelu tarjoaa helpon tavan monistaa elementtejä, jos sama elementti toistuu jokaisella tai usealla asiakirjan sivulla samassa kohtaa, kuten ylä- tai alatunniste. Elementeille voidaan myös asettaa taustaväri, reunat ja läpinäkyvyys. Kuviossa 4 esitellään kaikille elementeille yhteisiä perusasetuksia.



KUVIO 4. Perusasetukset

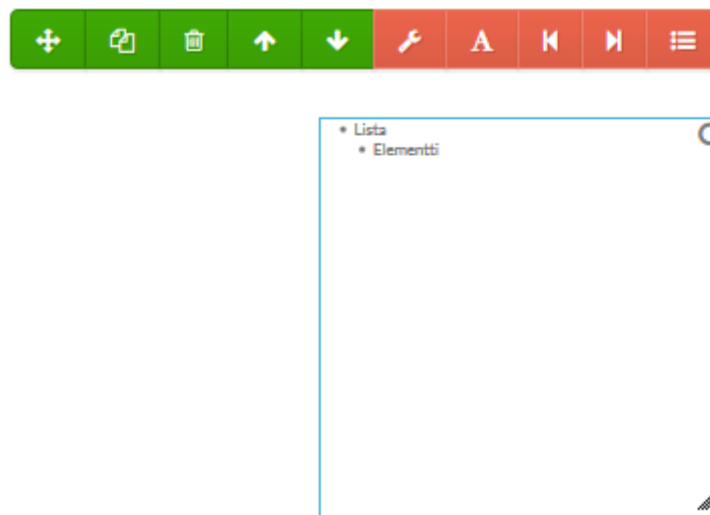
Työkalu tarjoaa kolme erilaista tekstielementtiä. Tekstielementtien asetuksista löytyy fontin valinta, tekstin väri, fonttikoko, tasaus, lihavointi, kursivointi ja alleviivaus. Kuviossa 5 esitellään tekstielementtien yhteisiä tekstiasetuksia. Tekstikenttä mahdollistaa käyttäjän kirjoittaa tavallista tekstiä asiakirjaan.



KUVIO 5. Tekstiasetukset

Lista mahdollistaa sekä järjestettyjen ja ei-järjestettyjen listojen luonnin.

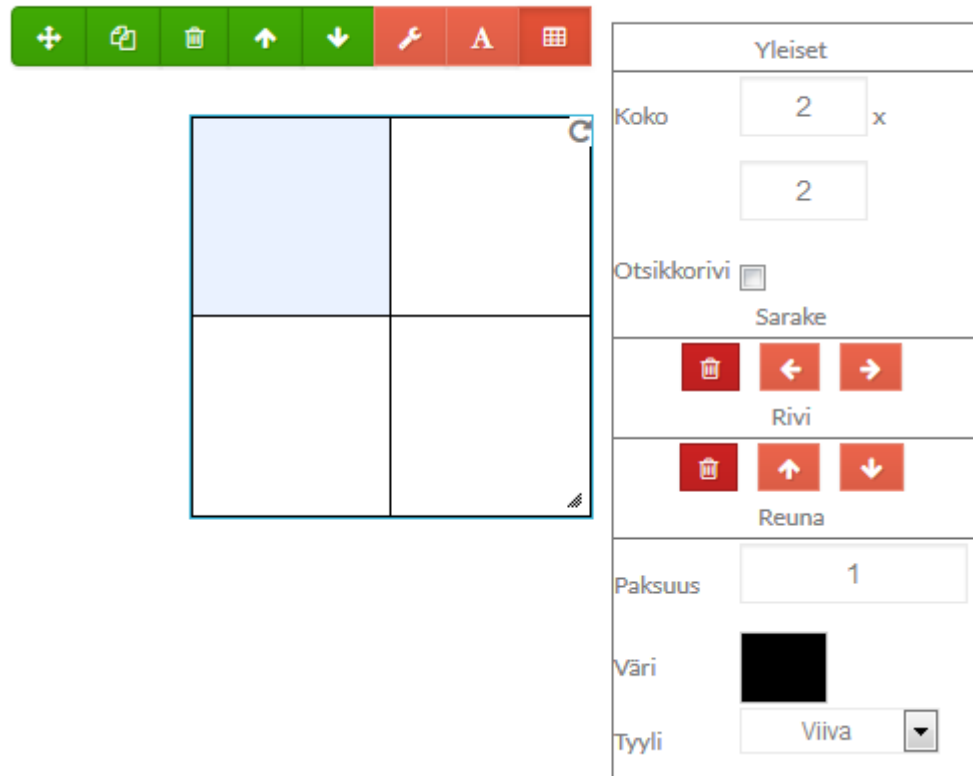
Listaelementti mahdollistaa monitasoisen listojen luonnin. Listaelementin työkaluista löytyvät näppäimet listan tasojen ja tyyppin muokkaamiseen. Kuviossa 6 esitellään listan työkaluja.



KUVIO 6. Listaelementin työkalut

Taulukkoelementti mahdollistaa erilaisten taulukoiden helpon luonnin. Työkalu tarjoaa mahdollisuuden lisätä ja poistaa sarakkeita sekä rivejä helposti.

Taulukkoon on mahdollista lisätä otsikkorivi. Taulukon ulkoasua voidaan muuttaa taulukkotyökalusta. Kuvassa 7 esitellään taulukkoelementin työkalu.



KUVIO 7. Taulukkoelementin työkalu

Työkalu tarjoaa asiakirjojen luonnissa tarvittavia rakenne-elementtejä. Rakenne-elementeistä löytyvät automaattinen sisällysluettelo ja sivunumerointi. Mikä tahansa elementti voidaan määrittää otsikoksi, joka näkyy sisällysluettelossa automaattisesti. Työkalu mahdollistaa ensimmäisen sivun sivunumeron määrittämisen. Kun sivunumero määritellään, työkalu päivittää kaikki sivunumerot ja sisällysluettelot automaattisesti vastaamaan käyttäjän syöttämää ensimmäistä sivua.

Työkalu tarjoaa kolme erilaista rakennuselementtiä, joilla voidaan luoda helposti asiakirjoihin erilaisia grafiikoita. Työkalussa on käytettävissä nelikulmio, kolmio ja ellipsi, joiden kokoa ja väriä voidaan muuttaa.

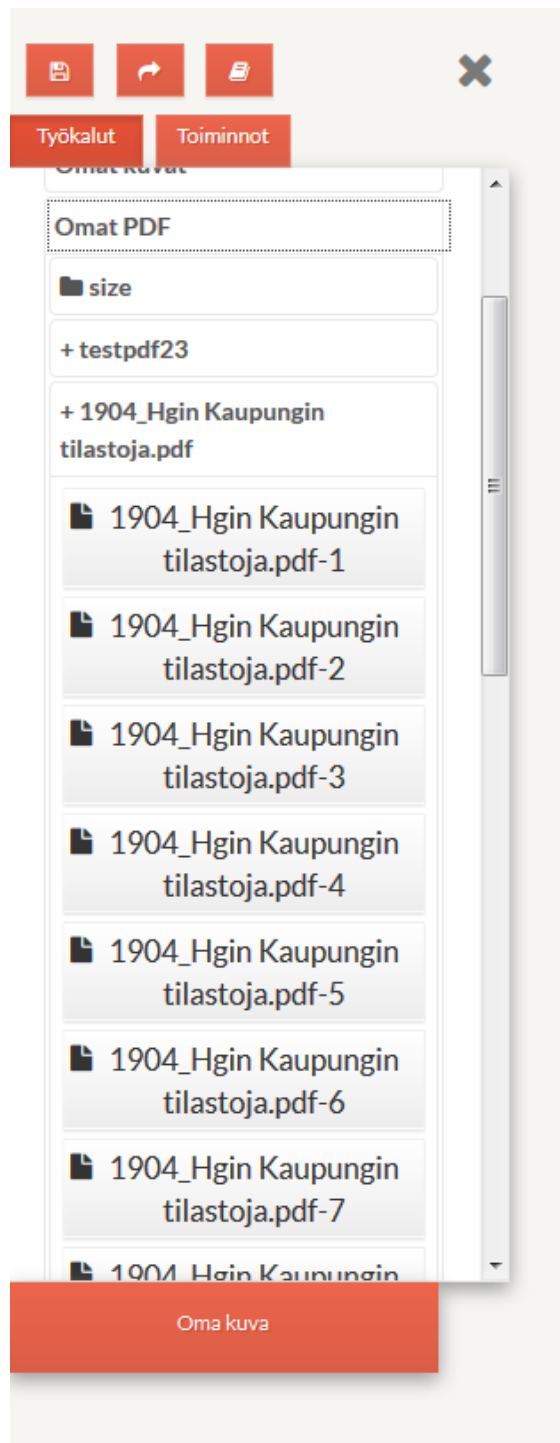
Työkalu tarjoaa myös kokoelman erilaisia taloyhtiöiden yleisesti käyttämiä sivupohjia, joita voidaan raahata asiakirjaan. Työkalusta löytyy valmiina

esitetyt kunnossapitosuunnitelmataulukko, korjausohjelmataulukko, korjausohjelmataulukko, yhtiövastiketaulukko ja kunnossapitoehdotus.

Palveluun tuodut omat kuvat ja PDF:t löytyvät työkalusta niille varatuista kohdista. Työkalu näyttää yrityksen kansiorakenteen käyttäjän avattua omille kuville tai PDF:lle tarkoitetun valikon. Kun käyttäjä avaa omat kuvat, palvelu näyttää kansiodien sisällä vain kuvatyyppejä olevia tiedostoja. Omat PDF:t -valikon sisällä työkalu näyttää vain PDF-tyypin tiedostoja.

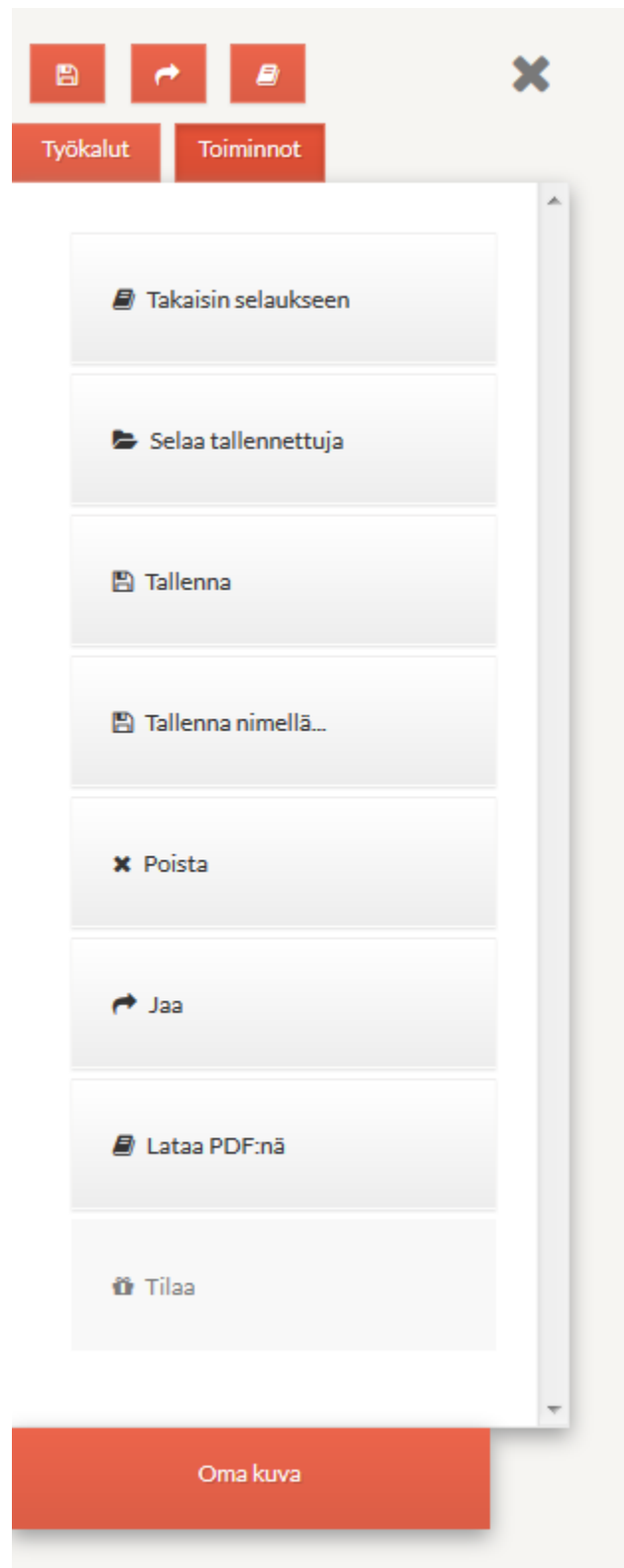
Kun käyttäjä lisää oman kuvan asiakirjaan, luo työkalu kuvasta pienemmän tarkkuuden omaavan version, jonka se renderöi näytölle. Kuvasta luodaan pienemmän tarkkuuden versio ohjelmiston keventämiseksi. Asiakirjat, joissa on paljon alkuperäillä tarkkuuden olevia kuvia, ovat raiskaita muokata selaimelle.

Jos PDF:n tuonnin yhteydessä on määritelty useampi sivu tuotavaksi, muodostaa palvelu PDF:n sivuista paketin. Kuviossa 8 esitellään palvelun muodostamaa PDF-paketti. Käyttäjä voi lisätä kokonaisia PDF-paketteja tai yksittäisiä sivuja asiakirjaan. Käyttäjä voi myös lisätä yksittäisen sivun PDF-paketin sisältä asiakirjaan. Jos käyttäjä raahaa kokonaisen PDF-paketin, lisää työkalu automaattisesti jokaisesta paketin sivusta uuden sivun asiakirjaan. Kun käyttäjä lisää yksittäisen PDF-sivun asiakirjaan, lisää työkalu sille uuden sivun. Lisätyillä sivuilla näkyvät aiemmin toistuvaksi määriteltyt elementit, mikä mahdollistaa omien sivunumeroiden, ylä- ja alatunnisteiden käytön myös PDF-sivuja sisältävillä sivuilla.



KUVIO 8. PDF-paketti

Työkalun toiminnoista löytyvät tallenna, tallenna nimellä, poista, jaa ja lataa PDF:nä. Jakamistoiminto mahdollistaa asiakirjan jakamisen helposti sähköpostitse oikealla kohdeyleisölle räätälöidyllä saateviestillä. Lataa PDF:nä -toiminto mahdollistaa asiakirjan lataamisen tiedostona esimerkiksi tulostamista varten. Kuviossa 9 esitellään työkalusta löytyvät toiminnot.



KUVIO 9. Työkalun toiminnot

5 WWW-TEKNOLOGIAT

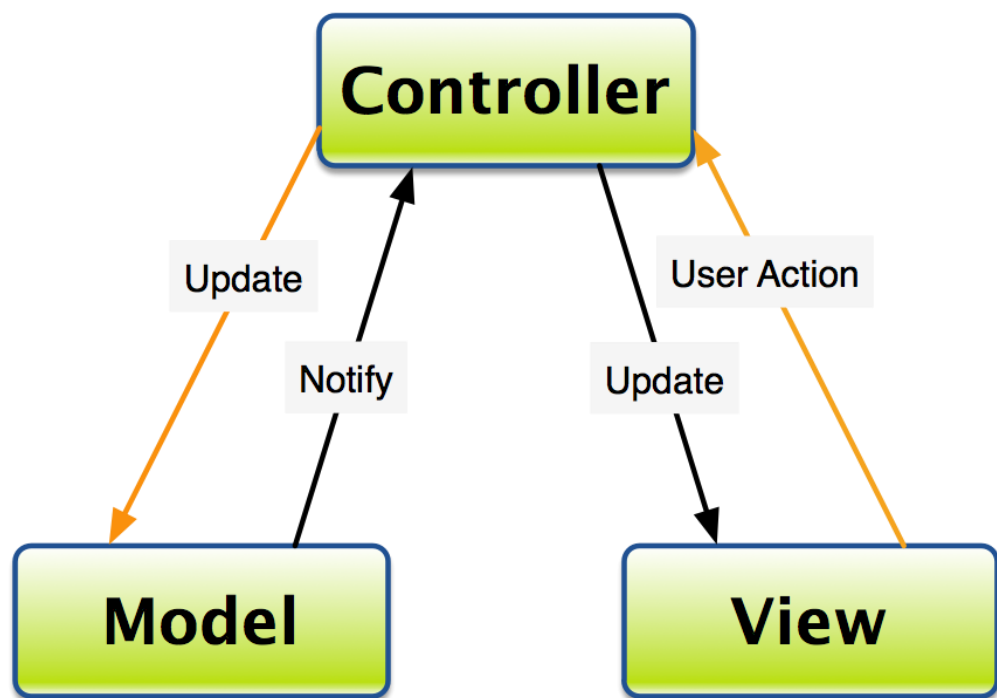
5.1 MVC-arkkitehtuuri

MVC (Model-View-Controller)-arkkitehtuuri on sovellusarkkitehtuurimalli, jonka päätarkoituksena on ratkaista päivitettävyyssongelmia. Useiden sähköisten palveluiden tarkoituksena on hakea tieto tietovarastosta ja esittää se käyttäjälle. Kun käyttäjä tekee muutoksen tietoon, tallentaa järjestelmä tehdyn muutoksen tietovarastoon. Keskeisin tietovirta tapahtuu tietovaraston ja käyttöliittymän välillä, jolloin koodin vähentämiseksi ja ohjelman nopeuttamiseksi tiedon haku, esittäminen ja muuttaminen tehtäisiin samaan tiedostoon. Tämä kuitenkin aiheuttaa ongelman päivitettävyydessä. Tietojärjestelmien käyttöliittymät vaihtuvat huomattavasti useammin kuin tietovarastohallintakoodi. Kun tällaisen ohjelman käyttöliittymä vaihdetaan, tietovarastohallintakoodin ollessa samassa tiedostossa täytyy sekin tehdä uudestaan. Toisena ongelmana yhteen tiedostoon laaditussa kokonaisuudessa ilmenee bisneslogiikan kasvaessa. Ongelmaksi syntyy tiedostokoon kasvaminen. Tiedostokokojen kasvaessa järjestelmän päivitettävyys heikkenee huomattavasti. (Microsoft 2015.)

MVC-arkkitehtuuri esittää ratkaisuksi päivitettävyyssongelmaan modulaarisuutta. Tietojärjestelmien kolme selkeää osa-aluetta jaetaan omiin tiedostoihin, jolloin yhden näistä osista vaihtuessa muihin ei tarvitse koskea. MVC-arkkitehtuuri nimensä mukaisesti jakaa tietojärjestelmät modeleihin, vieweihin ja controllereihin. Suomenkieliset vastineet näille ovat mallit, näkymät ja ohjaimet. Malli on vastuussa tietovaraston hallinnasta. Näkymä on vastuussa mallista tulevan tiedon esittämisestä. Ohjain on vastuussa näkymästä tulevien ohjaussyötteiden lukemisesta ja niiden uudelleenohjauksesta joko takaisin näkymään tai malliin tarvittavien muutosten tekemiseksi. (Microsoft 2015.)

MVC-arkkitehtuurin modulaarisuus mahdollistaa myös erilaisten näkymien nopean tuottamisen. Jos samasta tiedosta tarvitsee näyttää taulukko yhteen tarkoitukseen ja kaavio toiseen tarkoitukseen, tarvitsee vain luoda uusi näkymä, joka on yhteydessä samaan malliin. Tietojärjestelmän osittaminen osiin mahdollistaa helpon työnjaon näkymäkoodin ja bisneslogiikkakoodin ollessa hyvin erilaisia. Henkilö, joka on erikoistunut loppukäyttäjälle näytettävään

näkymän ulkoasuun suunnitteluun ja toteuttamiseen, voi työskennellä näkymän parissa, kun bisneslogiikkaan erikoistunut henkilö voi työskennellä ohjaimien ja mallien parissa. Harvemmin samalla henkilöllä on erikoisosaamista molemmilta osa-aloilta. Kuviossa 10 on esitetty Concrete5-sisällönhallintajärjestelmässä käytettävää MVC-sovellusarkkitehtuuria ja sen eri osien välisiä interkatioita. Käyttäjän näkymässä tekemät tapahtumat lähetetään ohjaimelle, joka niiden käsittelyn jälkeen tekee tarvittavat toimenpiteet tietokantamalleille. Mallit palauttavat tietoa ohjaimelle, joka käsittelee saamansa tiedon ja antaa sen näkymälle. Näkymä tulostaa tiedot käyttäjälle. (Microsoft 2015.)



KUVIO 10. MVC-sovellusarkkitehtuurimalli

5.2 HTML, CSS, PHP ja JavaScript

HTML (Hypertext Markup Language) on kuvauskieli, joka kertoo loppukäyttäjän käyttämälle internetselaimelle ladatun sivun rakenteen. HTML ei ota kantaa sivun ulkoasuun. HTML ei myöskään ole ohjelmointikieli. HTML-elementit erotetaan toisistaan tageilla. Tagit esiintyvät lähes aina pareittain alku- ja lopputagina. Haluttu tieto kirjoitetaan näiden tagien väliin. Koska HTML ei ota kantaa sivun

ulkoasuun, tarvitaan CSS (Cascading Style Sheets) tyyliohjekielä. CSS määrittelee sivun ulkoasun HTML-tagikohtaisesti. (W3schools 2015b.)

PHP (PHP: Hypertext Preprocessor) on avoimen lähdekoodin web-kehittämiseen erikoistunut palvelimen päässä ajettava ohjelmointikieli. PHP:llä on mahdollista luoda monimutkaisia web-sivuja ja web-sovelluksia. PHP:tä on mahdollista kirjoittaa HTML:n sekaan. PHP:llä on myös mahdollista tulostaa kokonaisia HTML-sivuja. (PHP 2015.)

JavaScript on komentosarjakieli, jonka päätehtävänä on lisätä web-sivujen ja web-sovellusten interaktiivisuutta. JavaScriptillä saadaan lisättyä dynaamisia toimintoja, kuten lomakkeen esitarkastaminen, web-sivun sisällön muutoksia ilman sivun uudelleenlataamista ja popup-ikkunoita. JavaScript on täysin asiakkaan päässä ajettava koodieieli. (W3schools 2015a.)

5.3 jQuery, jQuery UI ja AJAX

jQuery on nopea, pieni ja monipuolinen JavaScript-kirjasto. jQuery helpottaa monia asioita, kuten HTML-dokumentin läpikäyntiä ja muuttamista, tapahtumien hallintaa, animaatioita ja AJAX (Asynchronous JavaScript And XML)-kutsuja. jQuery-kirjastoa käyttävät miljoonat ihmiset. jQueryn suosion syy löytyy sen monipuolisuuden ja laajennattavuudesta. jQueryä käytetään interaktiivisten web-sivujen ja web-sovellusten luomisessa. (The jQuery Foundation 2015c.)

jQuery UI on kokoelma käyttöliittymään liittyviä vuorovaikutuksia, efektejä, vempaimia ja teemoja. jQuery UI on jQuery JavaScript -kirjaston lisäosa. jQuery UI tarjoaa interaktiivisempia rakennuselementtejä web-sivujen ja web-sovellusten kehitykseen. jQuery UI:n avulla on mahdollista luoda työpöytäsovelluksen kaltaisia web-sovelluksia. (The jQuery Foundation 2015b.)

Perinteisesti web-sivut vaativat uudelleenlatauksen, jotta niiden sisältö päivittyy. Sivun uudelleenlataus on hidasta ja kuormittaa palvelinta turhaan. Kun web-sivu uudelleenladataan, ladataan sivun tiedot kokonaan uudestaan. AJAX:n avulla on mahdollista ladata web-sivun tietyn osan tiedot uudestaan ja päivittää ne dynaamisesti ilman käyttäjän toimenpidettä. Tämä onnistuu helposti jQueryn tarjoamalla ajax-funktiolla. Kuviossa 11 esitellään jQueryn tarjoama ajax-funktio.

jQuery lähettää post-tyyppisen pyynnön palvelimelle, vastaanottaa palvelimelta vastauksen pyyntöön ja näyttää vastauksen ponnahdusikkunassa. (The jQuery Foundation 2015a.)

```
1 $.ajax({  
2   method: "POST",  
3   url: "some.php",  
4   data: { name: "John", location: "Boston" }  
5 })  
6 .done(function( msg ) {  
7   alert( "Data Saved: " + msg );  
8 });
```

KUVIO 11. JQueryn ajax-funktio

6 CONCRETE5-SISÄLLÖNHALLINTAJÄRJESTELMÄ

Concrete5 on sisällönhallintajärjestelmä, joka on toteutettu PHP:llä MVC-arkkitehtuuria käyttäen. Concreten ensimmäisen versio julkaistiin 2003.

Concreten kehittyessä viidenteen merkittävään päivitykseen se uudelleen nimettiin Concrete5:ksi ja julkaistiin MIT-lisenssillä avoimena lähdekoodina. Vuosien 2008 ja 2014 välillä on julkaistu 7 Concrete5-versiota. Concreten kehityksessä pääpaino on ollut aina käytettävyyden helppoudessa. Concrete5 tarjoaa helppokäyttöisen työkalun, jolla sivustoja voi rakentaa osaamatta koodata ollenkaan. Sivuston rakennustyökalu ei kuitenkaan hidasta sivustoa tai rajoita kehittyneiden käyttäjien oman koodin kehittämistä. (Concrete5 2015g.)

6.1 Käyttöönotto

Concrete5-sisällönhallintajärjestelmä vaatii palvelimelta PHP 5.3.3:n tai uudemman. Concrete5 asennetaan purkamalla ladattu zip-paketti palvelimen public_html-kansioon. Ennen asentamista tulee tarkistaa ovatko tiedosto-oikeudet kunnossa. Concrete5 vaatii kirjoitusoikeuden files-, config-, packages- ja updates-kansioihin. Concrete5:lle tulee valmistella MySQL-tietokanta ja tietokantakäyttäjä kaikilla oikeuksilla. Valmisteluiden jälkeen loppuasennus voidaan suorittaa selaimella navigoimalla sivuston URL (Uniform resource locator)-osoitteeseen. Asennus pyytää käyttäjää valitsemaan halutun kielen sivustolle, minkä jälkeen asennus on valmis. (Concrete5 2015c.)

6.2 Rakenne

Concrete5-sisällönhallintajärjestelmä käyttää MVC-arkkitehtuuria. Järjestelmän kansiorakenteesta löytyvät kansiot MVC-arkkitehtuurin tarvittaville osille. Mallit, näkymät ja ohjaimet on jaettu omiin kansioihin, jotka löytyvät järjestelmän juuresta. Mallit luodaan models-kansioon ja ohjaimet luodaan controllers-kansioon. Yksittäisten sivujen tapauksessa näkymät luodaan single_pages-kansioon. Toistuvien sivujen näkymät vastaavat teemakansion alta löytyviä sivutyypitiedostoja. Myös sivutyypeille voidaan luoda omia ohjaimia. Sivutyypien ohjaimet tulee luoda controllers-kansion sisällä olevaan page_types-

kansioon. Ohjaimen nimen ja tiedostopolun tulee vastata siihen osoittavaa näkymää nimeltä ja tiedostopolulta. (Concrete5 2015c.)

6.3 Ohjaimet

Concrete5-sisällönhallintajärjestelmän sivut käyttävät taustallaan ohjaimia. Ohjaimet ovat yksinkertaisesti automaattisesti ajettavia koodia sivun renderöinnin yhteydessä. View-funktioita kutsutaan, jos mitään muuta ohjaimen funktioita ei ole täsmällisesti kutsuttu. Kuviossa 12 esitellään ohjaimen view-funktion URL-parametrejä, mallien ja set-funktion käyttöä. On_start-funktioita kutsutaan aivan ensimmäiseksi sivun renderöintiprosessissa. On_before_render-funktio kutsutaan ohjaimen funktioista viimeisenä aivan juuri ennen kuin ohjain luovuttaa renderöintiprosessin hallinnan näkymälle. Jos näkymässä ei ole määriteltä erikseen, mitä renderöintifunktioita halutaan käyttää, ajetaan ohjaimessa ensiksi on_start, jonka jälkeen ajetaan oletusarvoinen view-funktio ja lopuksi ajetaan on_before-render. (Concrete5 2015c.)

```

3 public function view($id=0) {
4     Loader::model('page');
5     $page = new Page($id);
6     $this->set('view', $page->getContent());
7 }

```

KUVIO 12. View-funktio

Aiemmin mainittujen renderöintiprosessin eri vaiheiden funktioiden sisällä voidaan käyttää muita ohjaimen tarjoamia MVC-arkkitehtuuriin liittyviä funktioita. Ohjaimessa voidaan lisätä näkymän ylätunnisteeseen koodia käyttämällä addHeaderItem-funktiota, joka ottaa vastaan yhden string-muotoisen parametrin, joka kirjoitetaan näkymän ylätunnisteeseen sellaisenaan. Post-funktiolla voidaan palauttaa näkymästä tulleiden post-viestien arvoja antamalla sille halutun post-viestin avaimen. Post-viestistä saatuja arvoja voidaan käyttää määrittelemään,40se mitä halutaan renderöidä ohjelmalogiikan avulla. Ohjaimessa käytettävällä set-funktioilla voidaan viedä muuttujia helposti ohjaimesta näkymään. Set-funktio ottaa ensimmäisenä parametrina näkymässä käytettävän muuttujan nimen ja toisena parametrina sen halutun arvon. Näkymässä ohjaimen

set-funktiolla asetettuja muuttujia voidaan kutsua ilman alustamista näkymän päässä. (Concrete5 2015c.)

6.4 Mallit

Concrete5-sisällönhallintajärjestelmä ei pakota käyttämään perinteisiä malleja vaan suosittelee luomaan tietokantoja hallinnoivat luokat siten, että yksi luokka käsittelee pääsääntöisesti yhtä tietokantataulua. Sisällönhallintajärjestelmän käyttämät mallit tulee luoda models-nimiseen kansioon, joka löytyy järjestelmän juuresta. Kuviossa 13 esitellään Concrete5-malli. (Concrete5 2015a.)

```

1  <?php
2  defined('C5_EXECUTE') or die(_("Access Denied."));
3
4  class Example {
5      public static function getAll() {
6          Loader::db();
7          $sql = "select * from Examples";
8          $result = $db->GetAll($sql);
9          return $result;
10     }
11 }
12

```

KUVIO 13. Concrete5-malli

6.5 Näkymät

Concrete5-sisällönhallintajärjestelmän näkymät ovat vastuussa loppukäyttäjälle näytettävän tiedon tulostamisesta. Näkymiä on järjestelmässä kahdenlaisia. Järjestelmään voidaan luoda sivutyyppejä, jotka mahdollistavat useiden sivujen luomisen käyttäen samaa näkymää. Toinen vaihtoehto on luoda yksittäinen sivu. (Concrete5 2015c.)

6.5.1 Yksittäiset sivut

Yksittäinen sivu on sivustolla näkyvä sivu, joka ei toistu. Yleisesti käytettävä yksittäinen sivu on kirjautumissivu. Kirjautumissivuja ei tarvita kuin yksi kappale sivustolla, joten se toteutetaan käyttäen yksittäistä sivua. Yksittäisten sivujen

taustakoodi usein käyttää hyödyksi sivun ainutkertaisuutta. Taustakoodin ei tarvitse tarkistaa, mikä sivun instanssi on kyseessä, vaan aina voidaan olettaa sen olevan sama. Yksittäisten sivujen ollessa ainutlaatuisia sivustolla ei niille luoda omaa sivutyyppiä oletusarvoisesti sivuston käyttämään teemaan. Yksittäiset sivut käyttävät oletusarvoisesti kaikille yksittäisille sivuille yhteiseksi tarkoitettua sivutyyppiä View. Tarvittaessa yksittäiselle sivulle voidaan luoda oma sivutyyppi käytössä olevan teeman kansion juureen käyttäen kyseessä olevan yksittäisen sivun nimeä. (Concrete5 2015e.)

Yksittäiset sivut tehdään luomalla PHP-tiedosto sivuston tiedostonhallinnan juuresta löytyvään single_pages-kansioon. Single_pages-kansioon luotavien PHP-tiedostojen ei tarvitse sijaita juuressa vaan niitä voidaan kansioida. Kansioiden nimet tulevat näkymään yksittäisen sivun lopullisessa osoitteessa, joka mahdollistaa sivujen ryhmittelemisen yhteisen osoittepolun alle. Kun PHP-tiedosto on luotu, pitää se vielä aktivoida käyttäen Concrete5:n hallintapaneelia. Yksittäiset sivut aktivoidaan hallintapaneelistä Sivut ja teemat alta löytyvästä Yksittäiset sivut-kohdasta. Hallintapaneeli pyytää ylläpitäjää syöttämään yksittäisen sivun osoiteen, joka on luodun tiedoston tiedostonimi ja kansiopolku ilman tiedostopäätettä. Kuviossa 14 esitellään yksittäisen sivun lisäämistä. Ylläpitäjän aktivoitua yksittäisen sivun, voidaan sille navigoida lisäämällä sivun osoite sivuston pohjaosoitteeseen kauttaviivan jälkeen esimerkiksi http://sivustosiosoite.fi/mahdollinen_kansio/yksittainen_sivu. (Concrete5 2015e.)

Lisää yksittäinen sivu

Sivu, jonka haluat lisätä on saatavilla osoitteessa:

Lisää

KUVIO 14. Yksittäisen sivun lisääminen

6.5.2 Sivutypit

Sivutyyppi on sivustolla toistuvien sivujen pohjana käytettävä tiedosto. Sivutyyppi mahdollistaa helpon ja nopean tavan luoda monta sivua, joissa on samanlainen rakenne. Tällaisia toistuvia sivuja voisi olla esimerkiksi tavallinen

sisältötekstisivu, jossa yläreunassa on ylätunniste ja keskellä sivua paikka sisältötekstille. Ainoa muuttuva asia näillä sivuilla olisi sisältöteksti.

Sivutyypit luodaan käytettävänä olevan teeman juureen. Luodut sivutyypit täytyy aktivoida sisällönhallintajärjestelmän hallintapaneelisti Sivut ja teemat alta löytyvästä Sivutyypit-kohdasta. Järjestelmä pyytää kuvaavan nimen sivutyypille sekä nimikkeen, joka tulee olla sama kuin aiemmin luodun sivutyypin tiedoston nimi teemakansion juuressa ilman tiedostopäätettä. Kuviossa 15 esitellään sivutyypin lisäystä. (Concrete5 2015c.)

KUVIO 15. Sivutyypin lisäys

6.6 Teemat

Teemat ovat vastuussa sivuston visuaalisesta ilmeestä. Teemat ovat kokoelma HTML-, CSS- ja PHP-tiedostoja. Teemakansioon voidaan myös tallentaa kyseisessä teemassa käytettäviä kuvaelementtejä, joita käytetään sivuston renderöimisessä loppukäyttäjän nettiselaimen. Teemat tukevat modulaarista sivurakennetta ylä- ja alatunnisteiden osalta. Ylä- ja alatunnisteet voidaan luoda

erikseen elements kansioon, jolloin niitä voidaan yhteisesti käyttää kaikissa halutuissa sivutyypeissä. (Concrete5 2015g.)

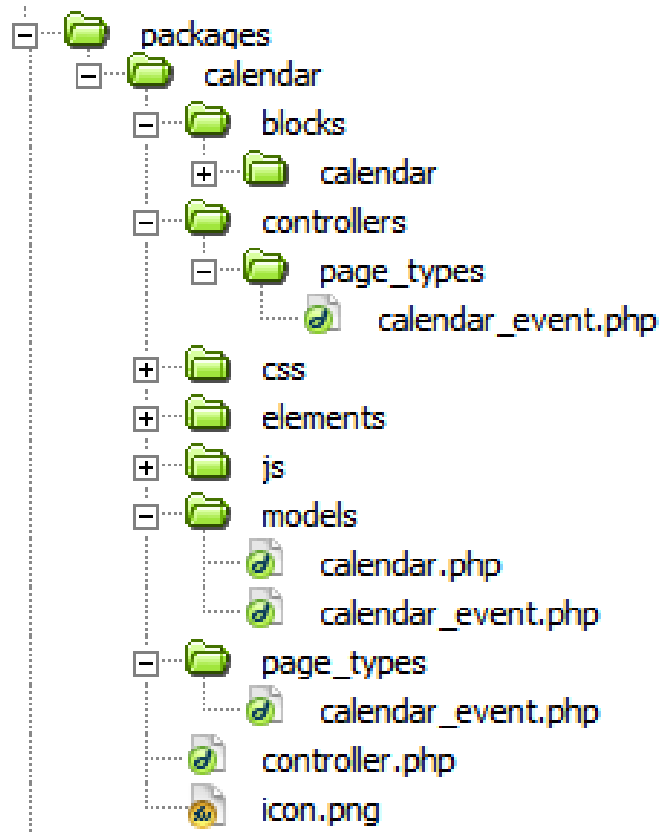
6.7 Työkalut

Työkalutiedostoja käytetään, kun sivuston tarvitsee vastata URL-kutsuun, mutta sivuston ei tarvitse renderöidä loppukäyttäjälle mitään visuaalista vastauksena. Yleinen käyttötapa työkaluille on AJAX-kutsulla kutsuttava rajapintapalvelu. Työkalut sijaitsevat sivuston juuresta tools-kansiossa. (Concrete5 2014.)

6.8 Paketit

Paketit ovat Concrete5-sisällönhallintajärjestelmän tarjoama tapa paketoita sivustolle luotavia erikoisominaisuuksia. Paketit ovat helppo jakaa, asentaa ja poistaa järjestelmästä. Paketteja voi ladata Concrete5:n kauppapaikasta. Pakettien asennus ei vaadi välttämättä järjestelmän yhdistämistä Concrete5:n verkkoon. (Concrete5 2015d.)

Pakettien sisäinen kansiorakenne mukailee Concrete5-sisällönhallintajärjestelmän omaa kansiorakennetta, mikä mahdollistaa järjestelmän tarjoamien funktioiden ja luokkien täydellisen ylikirjoittamisen. Pakettien tavoite on mahdollistaa niiden helppo uudelleenkäytettävyys ja siirrettävyys. Paketti voidaan kopioida toisen sivuston packages kansioon, minkä jälkeen sen asennus onnistuu helposti hallintapaneelin kautta. Esimerkkinä uudelleenkäytettävästä paketista voisi olla kalenterilmoduuli sivustolle. Kuviossa 16 esitellään paketin kansiorakennetta. (Concrete5 2015d.)



KUVIO 16. Paketin kansiorakenne

Paketin omassa juuressa tulee sijaita controller.php-tiedosto, joka sisältää perustietoja paketista, kuten sen tiedostokahvan, joka tulee olla sama kuin paketin kansion nimi, getPackageName-funktion, joka palauttaa paketin nimen, getPackageDescription-funktion, joka palauttaa paketin kuvauksen ja install-funktion, joka ajetaan paketin asennuksen yhteydessä. Install-funktion sisällä voidaan asentaa paketin sisältämiä asioita järjestelmään, kuten sivutyyppejä, teemoja ja yksittäisiä sivuja. (Concrete5 2015d.)

7 PALVELUN TOTEUTUS

7.1 Vaatimusten kartoitus

Palvelun vaatimuksia kartoitettiin projektin alkuvaiheessa projektiin osallistuvien osapuolien toimesta. Osapuolet kirjoittivat ylös omia sekä heidän tuntemien ihmisten tietämyksiä urakoitsijoiden ja talonyhtiöiden asiakirjojen laadintaan ja hallintaan liittyvistä tarpeista. Potentiaalisten asiakkaiden haastatteluissa ilmi tulleita mielipiteitä ja neuvoja kuunneltiin. Projektin edetessä projektiin saatiin lähtemään mukaan kiinnostuneita asiakkaita pilotoimaan palvelua.

Pilottiasiakkaiden käyttökokemukset ja kehitysideat on otettu huomioon palvelua kehitettäessä. Palvelua on esitelty myös Kiinteistöliitolle, josta saatiin kehitysideoita ja suuntaa antavia ohjeita palvelun kehittämiseen. Keskeisemmäksi vaatimukseksi nousi helppokäyttöinen keskitetty sähköinen vaihtoehto erilaisten asiakirjojen ja tiedotteiden laadintaan, arkistointiin ja jakamiseen asiakkaiden ja sidosryhmien välillä.

7.2 Yksittäiset sivut

Palvelu koostuu kolmesta yksittäisestä sivusta. Työkalusivu on vastuussa yksittäisten asiakirjojen lukemisesta ja muokkaamisesta. Työkalusivu on näkyvissä kaikille käyttäjille. Hallintasivu on vastuussa yritysten tietojen, käyttäjäryhmien ja käyttäjien ylläpitämisestä. Hallintasivu on näkyvissä vain käyttäjille, jotka ovat aktivoineet yhden neljästä eri käyttäjätasosta tai ovat saaneet ylläpitäjäoikeudet johonkin yritykseen. Rekisteröintisivu on vastuussa käyttäjien itsenäisestä rekisteröimisestä palveluun. Rekisteröintisivu on tarkoitettu palvelusta kiinnostuneille käyttäjille, jotka haluavat luoda omia yrityksiä palveluun. Käyttäjien, jotka vain lukevat tai muokkaavat jo olemassa olevan yrityksen asiakirjoja, ei tarvitse rekisteröityä itse. Yrityksen ylläpitäjä voi kutsua heidät palveluun hallintasivun kautta, jolloin kutsutuille käyttäjille luodaan automaattisesti käyttäjätili palveluun ylläpitäjän syöttämällä sähköpostiosoitteella.

7.3 Ohjaimet

Jokaisella yksittäisellä sivulla on oma ohjain. Ohjain on vastuussa osoiterivin lukemisesta ja sen tarjoaman tiedon perusteella näytettävän tiedon päättämisestä. Työkalu- ja hallintasivu sisältää monivaiheisen navigoinnin käyttäjän valitessa halutun yrityksen ja halutun toiminnon. Käyttäjän tekemät navigointivalinnat siirtyvät näkymästä ohjaimeen osoiterivin avulla.

Ohjaimet sisältävät näkymän tulostukseen liittyvien funktioiden lisäksi myös muita funktioita. Ohjaimien funktioita kutsutaan näkymistä POST-viestillä. Työkalun ohjaimesta löytyy funktioita muokattavan asiakirjan tallentamiseen, poistamiseen ja jakamiseen. Hallintasivun ohjaimesta löytyy funktioita asiakirjojen luonnissa käytettävien elementtien tuomiseen ja poistamiseen sekä käyttäjäryhmien ja käyttäjien lisäämiseen, muokkaamiseen ja poistamiseen. Rekisteröintisivun ohjain sisältää funktiot rekisteröinnin tekemiseen ja validointiin.

7.4 Mallit

Mallit vastaavat tietokannan hallinnoimisesta. Mallit hakevat, muuttavat ja lisäävät tietoa tietokantaan. palvelun toteutuksesta löytyy viisi mallia. Palvelusta löytyvät admin-, company-, document-, element- ja tooluser-mallit.

Admin-malli on vastuussa hallintasivulla käytettävien tietokantataulujen hallinnoimisesta. Admin-mallin __construct-funktiossa haetaan käyttäjän saamat oikeudet ja metatietoa yrityksistä, joissa hän on ylläpitäjänä. Mallista löytyy funktioita yrityksen tietojen, käyttäjäryhmien ja käyttäjien lisäämiseen, muokkaamiseen ja poistamiseen. Mallista löytyy funktioita käyttäjien oikeuksien tarkistukseen ja käyttäjätason määrittämien rajoitusten tarkistamiseen.

Company-malli on vastuussa yksittäisen yrityksen tietokantataulun hallinnoimisesta. __construct-funktiossa haetaan yrityksen tiedot, kuten nimi, y-tunnus, osoitetiedot ja perustajakäyttäjä. __construct-funktio voidaan määrittää hakemaan myös yritykseen liittyvät käyttäjäryhmät ja käyttäjät boolean tyyppisellä muuttujalla. Mallista löytyy funktioita yrityksen tietojen hakemiseen, muokkaamiseen ja tallentamiseen.

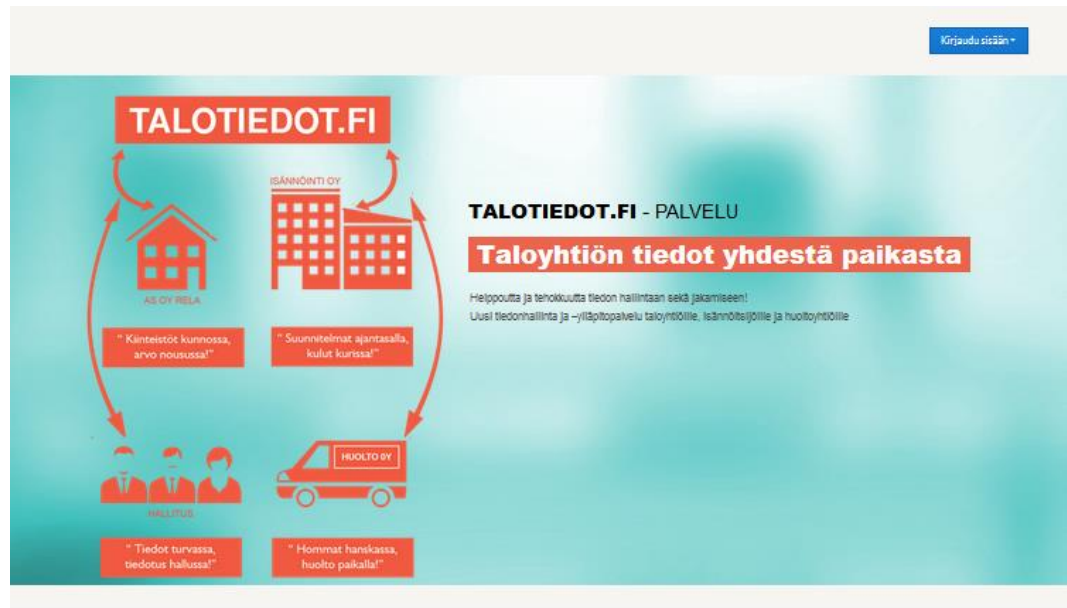
Document-malli on vastuussa yksittäisen asiakirjan tietokantataulun hallinnoimisesta. Syöttämällä `__construct`-funktioon asiakirjan tunnistenumero saadaan malli lataamaan asiakirjaan liittyvä tiedot tietokannasta. Jos tunnistetta ei syötetä funktioon, malli alustaa tyhjän asiakirjan ilman asetuksia. Asiakirjan asetuksista löytyvät asiakirjan nimi, tyyppi, yrityksen tunniste, luojakäyttäjän tunniste, sallitut käyttäjäryhmät ja asiakirjan sisältö. Mallista löytyy funktioita näiden asetusten hallintaa varten.

Element-malli on vastuussa yksittäisen asiakirjojen luonnissa käytettävän elementin tietokantataulun hallinnoimisesta. `__construct`-funktiossa haetaan elementin tiedot ja missä yrityksen kansioista se sijaitsee. Elementin tietoihin kuuluu elementin nimi, yrityksen tunniste, luojakäyttäjän tunniste, Concrete5 tiedostohallintajärjestelmään tallennetun kuvatiedostoon osoittava tunniste, elementin yksityisyysasetus, elementin tyyppi ja kansiotunniste. Mallista löytyy funktioita elementin tietojen hakemiseen, muuttamiseen ja poistamiseen.

Tooluser-malli on vastuussa työkalusivulla käytettävien tietokantataulujen hallinnoimisesta. `__construct`-funktiossa haetaan käyttäjän saamat oikeudet ja ryhmät joihin hän kuuluu. Mallista löytyy funktioita käyttäjän oikeuksien tarkistamiseen yrityksessä ja asiakirjoissa sekä yrityksen tietojen hakemiseen.

7.5 Teema

Palvelun teema on toteutettu Aplus HTML5 sivupohjan pohjalta. Palvelu etusivulla käytetään sivupohjan tarjoamaa sapluunaa, mutta palvelun muilla sivuilla käytetään räätälöityjä sivumalleja, joilla käytetään sivupohjan tarjoamia elementtejä ja tyylejä. Palvelun teemakansiosta löytyy kolme eri sivumallia. Home-sivumallia käytetään etusivulla, 1col-sivumallia käytetään palvelun toistuvilla sivuilla ja view-sivumallia käytetään yksittäisille sivuilla. Kuviossa 17 esitellään palvelun etusivua.



KUVIO 17. Etusivu

7.6 Työkalun toteutus

Työkalu on toteutettu käyttäen Concrete5:n yksittäisiä sivuja, ohjaimia, malleja, JavaScriptiä, jQueryä ja jQuery UI:ta. Näkymä lähettää URL:n avulla parametrina ohjaimelle halutun välilehden. Ohjain hakee tarvittavat tiedot käyttäen malleja ja palauttaa ne takaisin näkymälle set-funktiolla. Näkymä tulostaa oikein välilehden saatujen tietojen perusteella. Näkymän alatunnisteessa olevan JavaScript-tiedoston sisältämät funktiot ajetaan sivun renderöinnin jälkeen, jolloin niiden sisältämät HTML-elementtiinitykset aktivoituvat.

Työkalusivulle tullessaan palvelu listaa käyttäjälle kaikki yritykset, joihin hänellä on vähintään lukuoikeus. Käyttäjä valitsee halutun yrityksen listatuista yrityksistä, minkä jälkeen hänelle näytetään Windows-tiedostoselaimen kaltainen

kansiojärjestelmä, sisältäen ne kansiot ja tiedostot, joihin hänellä on vähintään lukuoikeus. Kansiojärjestelmässä käyttäjä voi lisätä, muokata ja poistaa kansioita sekä tiedostoja, jos hänelle on annettu yritykseen muokkaamisoikeus. Kaikki tiedostoihin luvat saaneet käyttäjät voivat selata tiedostoja lukutilassa, joka antaa mahdollisuuden vain lukea asiakirjaa tekemättä siihen mitään muutoksia. Painamalla muokkaa-näppäintä tiedostossa palvelu avaa varsinaisen työkalun. Työkalu mahdollistaa asiakirjojen luonnin ja muokkaamisen pilvipalveluna toteutulla tekstinkäsittelyohjelmalla.

Tekstinkäsittelyohjelma olessa toteutettu käyttäen JavaScriptiä, jQueryä ja jQuery UI:ta, kaikki asiakirjaan tehtävät muutokset tapahtuvat asiakkaan selaimessa. Asiakirjaa muokkaavat työkalut muokkaavat asiakirjan HTML-rakenteita ja CSS-tyylejä. Muutokset näkyvät reaaliajassa muokkaavalle käyttäjälle. Muutokset eivät tule voimaan kaikille käyttäjille ennenkuin muokkaaja on tallentanut asiakirjan. Tallennus tapahtuu muuttamalla koko asiakirjan sisältö base64-enkoodattuun merkkijonomuotoon jQueryn avulla. Base64-enkoodattu merkkijono lähetetään palvelimelle AJAX-kutsulla. Vastaanotettu base64 merkkijono tallennetaan tietokantaan käyttäen document-mallia.

Palveluun tuotavista PDF-tiedostojen sivuista luodaan kuvatiedostot käyttäen ImageMagic-ohjelmaa. PDF-sivuista muodostetaan kuvat niiden käytön mahdollistamiseksi työkalussa. Työkalu pystyy muokkaamaan HTML-rakennetta ja CSS-tyylejä. Kuvatiedostojen muodostamisessa PDF-sivusta käytetään 300 DPI (Dots per inch) tarkkuutta korkealaatuisen tulostamisen mahdollistamiseksi.

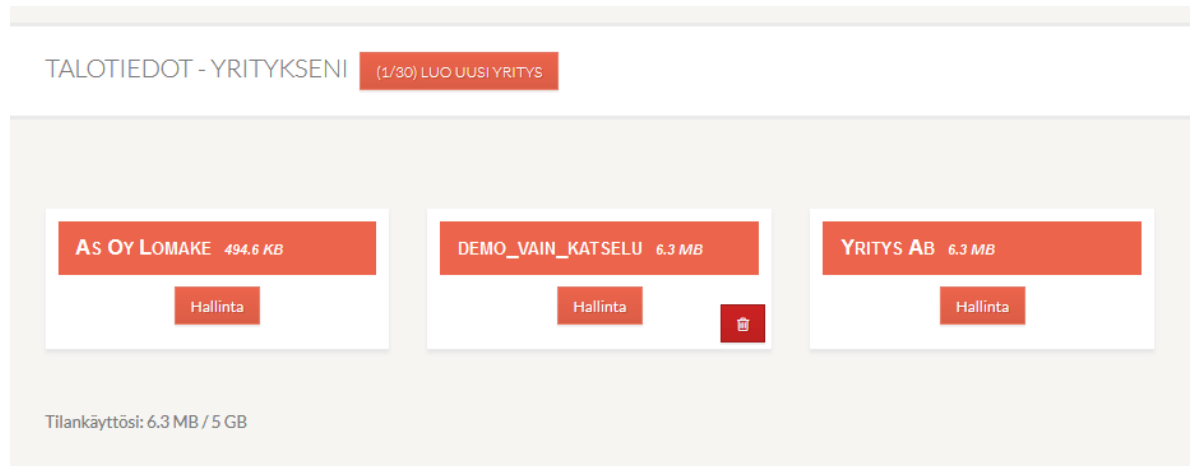
Kaikista kuvista, joita käytetään työkalussa, muodostetaan pienemmän kuvatarkkuuden versio käyttäen Concrete5 getThumbnail -funktiota. Pienen tarkkuuden versioita käytetään työkalussa, jossa kuvan tarkkuus ei ole tärkeää. Kuvan korkean kuvatarkkuuden version osoite tallennetaan kuvan HTML-elementtiin attribuuttina. Kun asiakirjasta muodostetaan PDF-tiedostoa jakamista ja tulostamista varten, haetaan kuvien korkean tarkkuuden osoitteet ja korvataan väliaikaisesti käytössä olevat pienemmän tarkkuuden osoitteet korkean tarkkuuden osoitteilla.

Tallennetusta asiakirjasta muodostetaan PDF-muotoinen tiedosto käyttäen webkittiin perustuvaa Wkhtmltopdf-kirjastoa. PDF-muotoinen tiedosto on saatavilla lataa-painikkeen kautta tai jakamalla asiakirja sähköpostitse palvelun avulla. Kun käyttäjä jakaa asiakirjan sähköpostitse, palvelu pyytää määrittelemään hallintasivun kautta määritellyistä käyttäjäryhmistä, kenelle sähköposti lähetetään. Jakamisen yhteydessä voidaan määrittää myös sähköpostiosoitteita erikseen. Jaettuun sähköpostiin voidaan määritellä oma teksti. Viestin lopussa on linkit asiakirjaan palvelun lukutilaan sekä aiemmin luotuun PDF-muotoiseen tiedostoon.

7.7 Hallintasivu toteutus

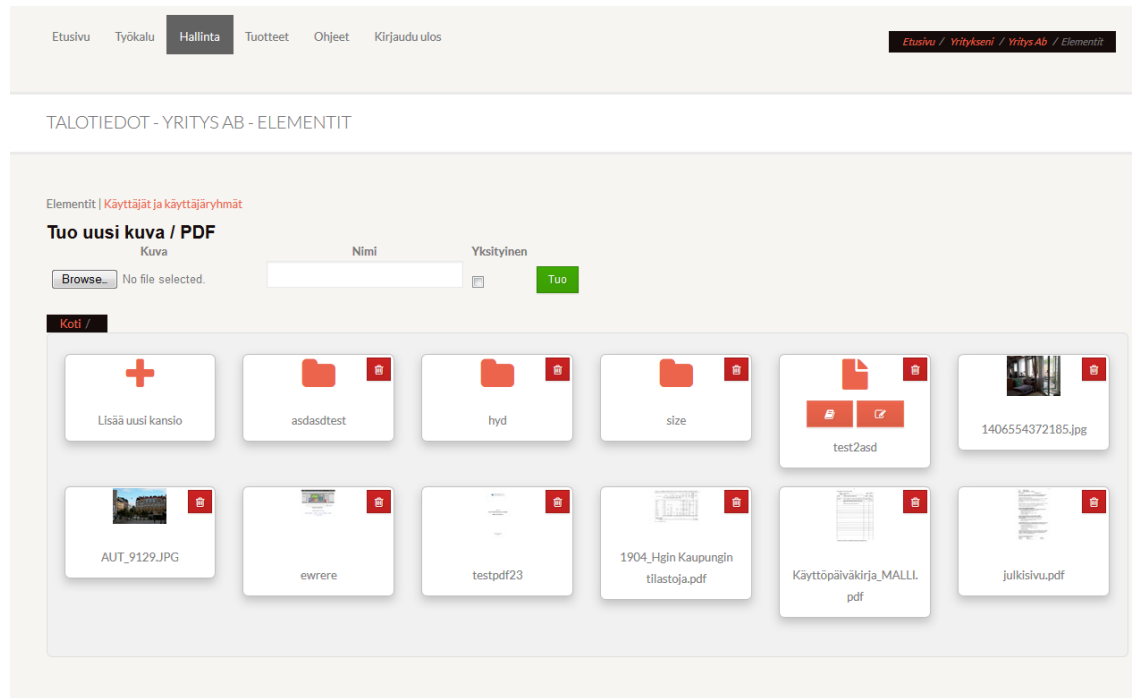
Kuten työkalu on hallintasivu toteutettu käyttäen Concrete5:n yksittäisiä sivuja, ohjaimia, malleja, JavaScriptiä, jQueryä ja jQuery UI:ta. Hallintasivun näkymä lähettää URL:n avulla parametrina ohjaimelle halutun välilehden. Ohjain hakee tarvittavat tiedot käyttäen malleja ja palauttaa ne takaisin näkymälle set-funktiolla. Hallintasivulla käytetään admin-mallia. Näkymä tulostaa oikein välilehden saatujen tietojen perusteella. Näkymän alatunnisteessa olevan JavaScript-tiedoston sisältämät funktiot ajetaan sivun renderöinnin jälkeen, jolloin niiden sisältämät HTML-elementtiinitykset aktivoituvat.

Käyttäjän saapuessa hallintasivulle hänellä näytetään lista yrityksistä joihin hänellä on ylläpitäjäoikeudet. Kuviossa 18 esitetään yrityslista. Jos käyttäjä on yrityksen perustajakäyttäjä, hänelle annetaan myös mahdollisuus poistaa kyseinen yritys ja kaikki sen tiedot. Käyttäjän valittuaan haluttu yritys näytetään hänellä kyseisen yrityksen hallintapaneeli. Elementit-välilehdellä käyttäjä näkee yrityksen koko kansiojärjestelmän. Kansiojärjestelmässä käyttäjä voi muokata ja poistaa kansioita ja yrityksen tiedostoja. Hallintapaneeli antaa mahdollisuuden käyttäjälle tuoda yrityksen kansioihin asiakirjojen luomisessa käytettäviä elementtejä kuten kuvia ja PDF-tiedostoja. PDF-tiedostoja tuodessa palvelu pyytää käyttäjää määrittelemään, mitkä sivut PDF:stä halutaan tuoda. Jos käyttäjä ei määritä haluttuja sivuja, palvelu lataa kaikki PDF:n sivut.



KUVIO 18. Yrityslista

Kansiojärjestelmä on toteutettu käyttäen JavaScriptiä, jQueryä ja jQuery UI:ta. Kuviossa 19 esitetään hallintasivun kansiojärjestelmä. Käyttäjä voi siirtää, uudelleen nimetä ja poistaa kansioita ja tiedostoja, joita on tuotu palveluun. Käyttäjä voi myös lisätä uusia kansioita. Käyttäjän tekemät muutokset lähetetään AJAX-kutsua käyttäen palvelimelle, joka tallentaa tehdyt muutokset tietokantaan. Kaikki käyttäjän tekemät muutokset tallentuvat reaaliajassa palvelimelle. Kansiodien sisältöjen lataus on myös toteutettu käyttäen AJAX-kutsuja ensimmäisen latausajan lyhentämiseksi. Kun käyttäjä navigoi kansiojärjestelmässä, järjestelmä seuraa käyttäjän kulkemaa polkua ja ylläpitää navigaatiopolkua. Navigaatiopolun avulla käyttäjä voi palata nopeasti takaisin aiemmalle kansiotasolle.



KUVIO 19. Hallintasivun kansiojärjestelmä

Hallintapaneelin Käyttäjät ja käyttäjärühmät -välilehdellä käyttäjä voi luoda uusia käyttäjärühmiä, määrittää eri käyttäjärühmien oikeuksia ja poistaa käyttäjärühmiä. Kuviossa 20 esitellään käyttäjärühmien hallintanäkymä. Määriteltäviä oikeuksia ovat ylläpitäjä- ja muokkaamisoikeudet. Kaikilla yrityksen käyttäjillä on automaattisesti lukuoikeudet heille jaettuihin asiakirjoihin. Käyttäjän luotuaan käyttäjärühmä, hän voi lisätä käyttäjärühmään käyttäjiä. Käyttäjien lisääminen tapahtuu syöttämällä lisättävän käyttäjän sähköpostiosoite ja valitsemalla haluttu käyttäjärühmä alasvetovalikosta. Lisätylle käyttäjälle lähtee sähköpostiviesti määritellyyn sähköpostiosoitteeseen, missä häntä pyydetään vahvistamaan käyttäjätilin luonti. Käyttäjiä on myös mahdollista poistaa yrityksestä. Poistetuille käyttäjille lähetetään poistotapahtumasta ilmoitus. Kuviossa 21 esitellään käyttäjien hallintanäkymä.

Etusivu
Työkalu
Hallinta
Tuotteet
Ohjeet
Kirjaudu ulos

Etusivu / Yrityksen / Yritys AB / Käyttäjät ja käyttäjäryhmät

TALOTIEDOT - YRITYS AB - KÄYTTÄJÄT JA KÄYTTÄJÄRYHMÄT

Elementit | Käyttäjät ja käyttäjäryhmät

Käyttäjäryhmät

Lisää käyttäjäryhmä

Nimi

Ylläpitäjä

Luontioikeudet

Lisää

Muokkaa käyttäjäryhmiä

Nimi	Ylläpitäjä	Luontioikeudet	
Isännöitsijä	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Poista
Hallitus	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Poista
Osakas	<input type="checkbox"/>	<input type="checkbox"/>	Poista

Tallenna muutokset

KUVIO 20. Käyttäjäryhmät

Käyttäjät

Lisää käyttäjä

Sähköposti

Käyttäjäryhmä

Lisää

Muokkaa käyttäjiä

Nimi	Sähköposti	Käyttäjäryhmä	
demo_hpj	demo@kunnossa.fi	Osakas	Poista
admin	tuki@koodiavain.fi	Osakas	Poista

Tallenna muutokset

KUVIO 21. Käyttäjät

7.8 Toteutuksen yhteenveto

Palvelun vaatimukset kartoitettiin yhdessä projektiin osallistujien ja potentiaalisten asiakkaiden kanssa. Kartoitettujen vaatimusten pohjalta suunniteltiin ja toteutettiin niitä vastaava palvelu. Palvelu on helppokäyttöinen keskitetty sähköinen vaihtoehto erilaisten asiakirjojen ja tiedotteiden laadintaan, arkistointiin ja jakamiseen asiakkaiden ja sidosryhmien välillä.

Toteutetussa palvelussa näkymät lähettävät URL-parametrina halutun toiminnon ohjaimille. Ohjaimet hakevat tarvittavat tiedot malleista ja palauttavat ne takaisin näkymälle set-funktiolla. Näkymä tulostaa tiedot käyttäjälle käyttäen teeman tarjoamia sivumalleja.

Kaikki käyttäjän tekemät muutokset asiakirjaan tapahtuvat vain asiakkaan selaimessa. Asiakirjoja muokataan työkalusivun tarjoamilla työkaluilla. Työkalusivun työkalut ovat toteutettu käyttäen jQueryä ja jQuery UI:ta. Työkalussa asiakirjoihin tehtävät muokkaukset tallennetaan palvelimelle lähettämällä base64-enkoodattu merkkijono asiakirjan sisällön HTML-rankenteista ja CSS-tyyleistä.

Hallintasivu mahdollistaa yrityksen tietojen ylläpitämisen. Hallintasivun elementit-välilehdellä voi ylläpitää palveluun tuotuja ja luotuja yrityksen tiedostoja. Käyttäjät ja käyttäjäryhmät välilehdellä voi ylläpitää yrityksen käyttäjien oikeuksia.

Osa palvelun interaktioista on toteutettu käyttäen AJAX-kutsuja jQueryllä sivujen uudelleenlatauksien vähentämiseksi. Käyttäjälle tarjottavat käyttöliittymäelementit on toteutettu käyttäen jQueryä ja jQuery UI:ta.

8 YHTEENVETO

Työn tavoitteena oli toteuttaa helppokäyttöinen palvelu, joka vastaa taloyhtiöiden kaikkiin asiakirjatarpeisiin. Taloyhtiöt tarvitsevat keskitetyn palvelun, jossa voidaan luoda, muokata, arkistoida ja jakaa asiakirjoja ja tietoa asukkaille ja sidosryhmille helposti.

Työn teoriaosuudessa käsiteltiin erilaisia web-teknologioita, joita käytettiin palvelun toteuttamiseen. Palvelussa käytettiin sekä palvelimen että asiakkaan teknologioita mahdollistamaan mahdollisimman sulava ja varma kokonaisuus. Asiakkaan pään teknologioita käytettiin luomaan helppokäyttöinen ja responsiivinen käyttöliittymä. Palvelimen pään teknologioita käytettiin mahdollistamaan tietoturvan eri osa-alueita, kuten saatavuutta, luottamuksellisuutta ja ajantasaisuutta.

Työn käytännön osuudessa käsiteltiin tarjottavaa palvelua ja sen toteutusta. Toteutuksen eri osa-alueet käsiteltiin yksityiskohtaisesti. Toteutuksen kuvaus vastaa palvelun nykytilannetta.

Palvelu on tällä hetkellä pilottivaiheen loppuhetkillä. Palvelun markkinointi on juuri alkamassa. Asiakkaiden tarpeita ja mielipiteitä on kuunneltu ja palvelua muutettu niiden perusteella. Palveluun halutaan ominaisuuksia, jotka helpottaisivat taloyhtiöiden toimimista aliurakoitsijoiden kanssa. Taloyhtiöiltä löytyy muitakin tiedotus- ja hallintatarpeita kuin asiakirjoihin liittyviä. Palvelua halutaan monipuoleistaa tulevaisuudessa tarjoamaan taloyhtiöille keskitetyn palvelun mahdollisimman moneen heidän tarpeisiinsa. Tulevaisuudessa palvelua on tarkoitus kehittää vastaamaan myös laajemman asiakaskentän tarpeita.

Jos joltain muulta asiakaskentältä huomataan puuttuvan vastaava palvelu, on tärkeää pitää palvelumme helposti ja nopeasti muokattavissa vastaamaan vastaavanlaisiin asiakirjatarpeisiin.

LÄHTEET

Concrete5. 2014. Getting the URL for a block or package tool [viitattu 16.3.2015]. Saatavissa: <http://www.concrete5.org/documentation/how-tos/developers/getting-the-url-for-a-block-or-package-tool/>

Concrete5. 2015a. Combining Views, Controllers, Events and Models to Create a Example Shopping Cart [Viitattu 16.3.2015]. Saatavissa: http://www.concrete5.org/documentation/how-tos/developers/shopping_cart_tutorial/

Concrete5. 2015b. System Requirements [viitattu 16.3.2015]. Saatavissa: <http://www.concrete5.org/documentation/developers/5.7/installation/system-requirements/>

Concrete5. 2015c. MVC approach [viitattu 16.3.2015]. Saatavissa: <https://www.concrete5.org/documentation/developers/5.6/pages/mvc-approach>

Concrete5. 2015d. Packages [viitattu 16.3.2015]. Saatavissa: <http://www.concrete5.org/documentation/developers/5.6/system/packages/>

Concrete5. 2015e. Single Pages [viitattu 16.3.2015]. Saatavissa: <https://www.concrete5.org/documentation/developers/5.6/pages/single-pages/>

Concrete5. 2015f. Themes [viitattu 16.3.2015]. Saatavissa: <http://www.concrete5.org/documentation/developers/5.6/pages/themes/>

Concrete5. 2015g. What is concrete5? [viitattu 16.3.2015]. Saatavissa: <http://www.concrete5.org/documentation/developers/5.7/background/what-is-concrete5/>

Microsoft. 2015. Model-View-Controller [viitattu 16.3.2015]. Saatavissa: <https://msdn.microsoft.com/en-us/library/ff649643.aspx>

PHP. 2015. What is PHP? [viitattu 16.3.2015]. Saatavissa: <http://php.net/manual/en/intro-what-is.php>

The jQuery Foundation. 2015a. Ajax [viitattu 16.3.2015]. Saatavissa:
<http://learn.jquery.com/ajax/>

The jQuery Foundation. 2015b. jQuery user interface [viitattu 16.3.2015].
Saatavissa: <http://jqueryui.com/>

The jQuery Foundation. 2015c. What is jQuery? [viitattu 16.3.2015]. Saatavissa:
<http://jquery.com/>

W3schools. 2015a. JavaScript Tutorial [viitattu 16.3.2015]. Saatavissa:
<http://www.w3schools.com/js/default.asp>

W3schools, 2015b. HTML(5) Tutorial [viitattu 16.3.2015]. Saatavissa:
<http://www.w3schools.com/html/default.asp>